

REMARKS

Claims 1-17 remain for consideration. The rejections of the claims are traversed in the arguments below.

The arguments presented in the Response dated January 19, 2004 are maintained in this response and incorporated by reference.

The asserted Gibbons-Shepard combination was further reviewed in view of the “Response to Arguments” made in the recent Office Action. The Office Action does not show that the Gibbons-Shepard combination teaches or suggests all the limitations of the claims.

In essence, the Gibbons-Shepard combination does not show or suggest a hierarchy of component object model interfaces and a nearly corresponding hierarchy of template classes. It is recognized that the “nearly corresponding hierarchy” language is not found in the claims. However, the associations of the first and second template classes with the component object model interfaces and the specified inheritance of the second template class from the first template class realize a nearly corresponding hierarchy. Specifically, in claim 1 there is a hierarchy of component object model interfaces, with the interface at a lowest level of the hierarchy inheriting from an interface at the highest level of the hierarchy. In addition, there are first and second template classes, the second inheriting from the first, and the first template class associated with the highest level interface in the hierarchy, and the second template class associated with the lowest level in the hierarchy. Further still, the limitations specify that the second template class, being associated with the lowest level in the interface hierarchy, is instantiated with an interface as a template parameter.

The Gibbons-Shepard combination does not suggest all of these limitations. First, the Gibbons-Shepard does not suggest the corresponding hierarchies of component object model interfaces and template classes. Second, the Gibbons-Shepard combination does not suggest the instantiation of the second template class (associated with the lowest hierarchy level) with an interface as a template parameter.

Gibbons generally teaches “interface inheritance”, and Shepard generally teaches “template inheritance.” However, these general teachings do not suggest all the specific limitations of the COM interface hierarchy and the corresponding (first and second) template

classes. The reasoning in the recent Office Action is apparently based on a single example from Shepard involving an interface being provided as a template parameter and Gibbons' general teaching of interface inheritance. These selected portions, however, do not suggest that first and second template classes are respectively associated with highest and lowest levels in the interface hierarchy. Specifically, the cited CApartmentOb is alleged to be a template class by virtue of the template parameters that follow. However, there is no suggestion that this template class is associated with the lowest level in an interface hierarchy. The general suggestion by Gibbons of an interface hierarchy does not imply that the alleged template from Shepard is associated with the lowest level in the interface hierarchy.

The limitations of claims 2-17 are not shown to be suggested as explained in the previous response.

The alleged motivation for combining the teachings of Gibbons and Shepard is insufficient to support a *prima facie* case of obviousness. The alleged motivation is to "allow for extending software flexibility and extensibility." This alleged motivation is conclusory and therefore improper. For example, the Office Action does not provide any evidence that either of Gibbons or Shepard lacks flexibility and extensibility. Nor does the Office Action identify those specific elements of the Gibbons system that could be modified with specific teachings from Shepard. Furthermore, the Office Action uses the general terms "flexibility and extensibility" without providing any definition of what these terms mean or how any purported combination would better achieve these objectives beyond what the single references achieve. Therefore, the alleged motivation is improper.


The Office Action fails to recognize the requirements to establish inherency. That is, the Office Action must show that the missing limitations are necessarily present in the reference. Inherency may not be established by showing that the limitations are possible in the cited reference. In regards to claim 2, it is respectfully submitted that the Office Action has only explained that direct inheritance is possible. The Office Action admits that the issue of direct inheritance is a function of how deep the classes are within a hierarchy. Thus, by implication the Office Action admits that inheritance other than direct is possible, and therefore, direct inheritance is not necessary. The other inherency-based rejections fail for similar reasons.

Withdrawal of the rejection and reconsideration of the claims are respectfully requested.
If the examiner has any questions or concerns, a telephone call to the undersigned is welcome.

No extension of time is believed to be necessary for consideration of this response.
However, if an extension of time is required, please consider this a petition for a sufficient
number of months for consideration of this response. If there are any additional fees in
connection with this response, please charge Deposit Account No. 50-0996 (USYS.007PA).

Respectfully submitted,

CRAWFORD MAUNU PLLC
1270 Northland Drive, Suite 390
Saint Paul, MN 55120
(651) 686-6633

By: 
Name: LeRoy D. Maunu
Reg. No.: 35,274